**Greatest Common Subset**

0086% of1124[mmalam](https://www.codewars.com/users/mmalam)

JavaScript

* [TRAIN AGAIN](https://www.codewars.com/kata/greatest-common-subset/train/javascript)
* [NEXT KATA](https://www.codewars.com/trainer/javascript)

Details

[Solutions](https://www.codewars.com/kata/greatest-common-subset/solutions/javascript)

[Discourse (11)](https://www.codewars.com/kata/greatest-common-subset/discuss/javascript)

* Add to Collection
* |
* Share this kata:

Write a function commonArray, that when given 2 arrays containing only integers, returns a sorted array that is the greatest common subset of the 2 arrays.

For Example:

if array1 is [1, 1, 2, 2, 3, 4, 5] and array2 is [1, 2, 2, 2, 3, 5], commonArray(array1, array2) should return an array containing the greatest common subset of the 2 arrays (with non-unique values included); so the output should be [1, 2, 2, 3, 5].

<https://www.codewars.com/kata/remember/train/csharp>

<script>

**const** commonArray = (array1, array2) => {

    da = {};

    db = {};

**for**(let i =0; i<array1.length; i++) {

        da[array1[i]] = 0;

   }

**for**(let i =0; i<array2.length; i++) {

        db[array2[i]] = 0;

   }

**for**(let i =0; i<array1.length; i++) {

        da[array1[i]]++;

   }

**for**(let i =0; i<array2.length; i++) {

        db[array2[i]]++;

   }

   ans = [];

**for**(**var** key **in** da) {

**if**(db.hasOwnProperty(key)) {

             let veces = Math.min(da[key], db[key]);

**for**(let j = 0; j < veces; j++) {

                ans.push( parseInt(key,10));

             }

        }

   }

**return** ans;

}

**var** arr1 = [2, 3, 3, 4, 5, 6, 6, 7];

**var** arr2 = [5, 6, 6, 6, 7, 8];

document.write(commonArray(arr1, arr2));

</script>